REMARKS

In response to the Office Action dated May 27, 2003, the Applicants respectfully request reconsideration. To further the prosecution of this application, amendments have been made in the claims, and the claims as presented are believed to be in allowable condition.

Introduction

The Office Action repeats the rejection of all pending claims under 35 U.S.C. §103 as purportedly being obvious over Ma in view Ha, applying the same analysis set forth in the prior Office Action. After receiving the previous Office Action, the Applicants' representatives conducted a telephone interview with the Examiner, in which the Examiner expressed concern about the breadth of claim 14. Subsequent to the interview, in a response filed February 26, 2003, claim 14 was amended to clearly distinguish over the subject matter which caused the Examiner concern (e.g., a file system having the capability of moving the location of data corresponding to a particular file from one volume to another). Nevertheless, the rejection has been repeated nearly verbatim in this Office Action, without meaningfully addressing how the limitations added to claim 14 are believed to be shown in the prior art. The Applicants will clarify the manner in which the claims distinguish below.

Rejections under 35 U.S.C. §102(e)

Claims 1, 7-8, 10-11, 34-38, 41, 43, 62, 69 and 71 are rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,167,408 to Cannon et al. ("Cannon"). This rejection is respectfully traversed.

Overview of the Present Invention

Many systems include a host computer and one or more storage systems that provide the host with logical volumes to store data, with the host computer perceiving the logical volumes as corresponding to physical storage devices (Applicants' specification, p. 1, lines 6-7 and p. 2, lines 10-14). Host computers may also include an application layer that refers to data objects with an identifier, such as a file name, which does not indicate where the file is physically stored among the logical volumes presented by the storage system (p. 2, lines 8-10). The host typically includes a logical volume manager (LVM) that maps each data object specified by the

application layer to particular logical volume locations (p. 2, lines 10-13). The identifier used by the LVM to refer to a logical volume is usually dependent on the physical configuration of the system, and may include information that describes a particular path through which the logical volume is accessible (p. 2, lines 14-23). These identifiers are assigned when the computer system is brought on line, and the host goes through a device discovery process (p. 3, lines 21-25).

Events can occur that may change a physical configuration of a computer system. For example, a port on the storage system or host may fail, and the system may need to be reconfigured so that one or more logical volumes are made available to the host over a different path (p. 4, lines 7-11). After the reconfiguration, the host computer conventionally needs to be rebooted to recognize the reconfigured nature of the system (p. 4, lines 11-13). After reconfiguration, the device identifier, which is assigned to one or more logical volumes and is dependent upon the physical configuration of the computer system, may change such that a logical volume may have a new identifier (p. 4, lines 13-19). Thus, when a physical reconfiguration takes place and those identifiers are altered, problems may arise for applications on the host (p. 4, lines 20-27). Specifically, after reconfiguration, a LVM may attempt to access logical volumes using the obsolete identifiers, which can result in the wrong data being accessed for a read or write operation (p. 5, lines 4-11).

Conventional computer systems do not provide any technique for automatically detecting changes in system configuration which can lead to the problems described above (p. 5, lines 12-14). To avoid such problems, a system administrator must manually determine a reconfiguration has occurred, and take action to avoid these problems. In accordance with one embodiment of the present invention, a technique is provided for automatically detecting changes in the system configuration, and for determining the manner in which the resources are to be accessed by the host computer in the new configuration. In addition, even if a system administrator detects that a reconfiguration has taken place, conventional systems do not provide for dynamically addressing such configuration changes. In accordance with another embodiment of the present invention, a technique is provided for dynamically reconfiguring a computer system, without reinitializing a host computer or application program thereon, to alter a manner in which an application accesses a system resource.

These aspects of the invention can be implemented in any of numerous ways, and are not limited to any particular implementation. In accordance with one embodiment, a host computer is provided with information that identifies available resources (e.g., logical volumes) in a manner independent of the physical configuration of the computer system. This information can then be correlated to information identifying the resources in a manner dependent on the physical configuration of the computer system. This mapping can be used to detect configuration changes, determine the manner in which resources are accessed according to a newly detected configuration, and dynamically reconfigure the system when a configuration change is detected.

The foregoing summary is provided to assist the Examiner in appreciating various aspects of the invention. However, this summary does not apply to each of the independent claims, and the language of the independent claims may differ in material respects from the summary provided above. Thus, Applicants respectfully request that careful consideration be given to the language of each of the independent claims and that each be addressed on its own merits, without relying on the summary provided above. In this respect, Applicants do not rely on the summary provided above to distinguish any of the claims of the present invention over the prior art. Rather, Applicants rely only upon the arguments provided below.

Claims 1-13

As amended, claim 1 recites a method, in a computer system including a host computer and at least one computer system resource accessible to the host computer, wherein the host computer comprises an operating system and the operating system uses at least one identifier to enable access by the host computer to the at least one computer system resource, of responding to changes in a configuration of the computer system impacting a manner in which the at least one computer system resource is accessed by the host computer. The method comprises steps of: (A) storing information relating to a first configuration of the computer system at a first point in time, the first configuration relating to a first manner of accessing the at least one computer system resource by the host computer, the first manner of accessing the at least one computer system resource including the use of a first identifier by the operating system; (B) determining a second configuration of the computer system at a second point in time, the second configuration relating to a second manner of accessing the at least one computer system resource by the host computer; (C) comparing the second configuration of the computer system with the first

configuration to determine whether the second configuration differs from the first configuration, in that the second manner of accessing the at least one computer system resource includes the use of a second identifier by the operating system, wherein the second identifier differs from the first identifier; and (D) when it is determined in the step (C) that the second configuration differs from the first configuration, determining the second identifier used by the operating system in the second manner of accessing the at least one computer system resource by the host computer.

In contrast, Cannon discloses a system for publishing operating parameters maintained by a "configuration manager" to one or more "managed units" (col. 2, lines 30-35). These parameters are stored on the managed units as "operating characteristics" in a profile (col. 2, lines 45-48). When an update to the parameters is to occur, the configuration manager compares a version code of each operating characteristic with a corresponding version code of a "reference characteristic" stored in a centrally maintained reference profile, and if the version codes do not match, the operating characteristic is replaced with the reference characteristic (col. 17, lines 14-22). Reference and operating characteristics include parameters such as the system administrator assigned to a managed unit, and when a backup of the managed unit is to be executed (Table 2).

First, Cannon does not disclose or suggest determining a second manner of accessing a computer system resource in response to a detected change in configuration. Instead, Cannon discloses comparing reference and operating characteristics based on a version code, and when the codes do not match, replacing the operating characteristic with the reference characteristic. Thus, even if the operating characteristics related to manners of accessing a resource, Cannon does not determine a second manner of accessing the computer system resource, but rather changes the second manner of accessing the resource in a predetermined way.

Second, Cannon does not disclose or suggest comparing two computer system configurations to determine whether they differ in the use of an identifier used by an operating system on a host computer to access at least one computer system resource. As discussed above, the sections of Cannon relied upon in the Office Action relate to characteristics which define system administration functions, and are entirely unrelated to determining changes to an identifier used by an operating system on a host computer to access to at least one computer system resource.

How are changes recognized?

Show or anticipates

For the reasons discussed above, claim 1 patentably distinguishes over Cannon, such that the rejection of claim 1 under 35 U.S.C. §102(e) as being anticipated by Cannon should be withdrawn.

Claims 2-13 depend from claim 1 and are patentable for at least the same reasons.

Claims 34-44

Independent claim 34 is directed to a computer-readable medium that, when executed on a host computer, performs a method substantially similar to the method recited in claim 1. Therefore, for the reasons set forth above with respect to claim 1, claim 34 patentably distinguishes over the prior art of record, such that the rejection of claim 34 under 35 U.S.C. §102(e) should be withdrawn.

Claims 35-44 depend from claim 34 and are patentable for at least the same reasons.

Claims 62-71

Independent claim 62 is directed to a host computer comprising, inter alia, an operating system for enabling access to the at least one computer system resource, the operating system using at least one identifier to enable access by the host computer to the at least one computer system resource; storing means for storing information relating to a first configuration of the computer system at a first point in time, the first configuration relating to a first manner of accessing the at least one computer system resource by the host computer, the first manner of accessing the at least one computer system resource including the use of a first identifier by the operating system; comparing means for comparing the second configuration of the computer system with the first configuration to determine whether the second configuration differs from the first configuration in that the second manner of accessing the at least one computer system resource includes the use of a second identifier by the operating system, wherein the second identifier differs from the first identifier; and second determining means for determining the second identifier used by the operating system in the second manner of accessing the at least one computer system resource by the host computer when it is determined by the first determining means that the second configuration differs from the first configuration.

As should be appreciated from the foregoing, Cannon fails to disclose comparing means for comparing first and second configurations, to determine whether an identifier used by the

operating system of the host computer to access at least one computer system resource differs. Thus, it is respectfully asserted that claim 62 patentably distinguishes over the prior art of record, such that the rejection of claim 62 under 35 U.S.C. §102(e) should be withdrawn.

Claims 63-71 depend from claim 62 and are patentable for at least the same reasons.

Rejections under 35 U.S.C. §103 Over Cannon in view of DeKoning

Independent claims 14, 45 and 72 are rejected under 35 U.S.C. §103 over Cannon in view of U.S. Patent No. 6,178,520 to DeKoning et al. ("DeKoning"). This rejection is respectfully traversed.

Claim 14 recites a method, in a computer system including a host computer and at least one computer system resource accessible to at least one application program executing on the host computer, wherein the host computer comprises an operating system and the operating system uses at least one identifier to enable access by the host computer to the at least one computer system resource, of reconfiguring the computer system. The method comprises a step of: (A) dynamically reconfiguring the computer system, without reinitializing the host computer or the application program, in response to a change in a configuration of the computer system that changes the at least one identifier used by the operating system to enable access by the host computer to the at least one computer system resource, to alter a manner in which the at least one application program accesses the at least one computer system resource.

The Office Action asserts that Cannon discloses the limitations of claim 14, with the exception of reconfiguring the computer system without reinitializing. The Office Action also asserts that DeKoning teaches reconfiguration without re-initialization, and concludes that it would have been obvious to a skilled artisan to incorporate the teachings of DeKoning into the disclosure of Cannon to ensure that "the system is not interrupted." These contentions are not supported by the references.

Without acceding to the propriety of the combination, claim 14 clearly distinguishes over the purported combination. As discussed above with reference to claim 1, Cannon does not disclose reconfiguring in response to a configuration change that results in a change in an identifier used by a host operating system to enable access by the host computer to at least one computer system resource. The Office Action asserts that this is shown by Table 2 of Cannon. In Table 2, Cannon discloses operating characteristics which define system administration

functions, such as the assignment of a system administrator, and says nothing about identifiers used to enable a host computer to access at least one computer system resource. Further, Cannon discloses imposing changes in those characteristics, and not responding to changes in the characteristics. As a result, Cannon does not disclose the type of configuration change recited in claim 14.

DeKoning also does not disclose or suggest dynamically reconfiguring a computer system in response to a change to at least one identifier used by the operating system to enable access by a host computer to at least one computer system resource as recited in claim 14. DeKoning teaches a system for recovering from disk drive insertion or removal in data storage systems (col. 2, lines 17-19). DeKoning teaches that drive insertion or removal can generate undesirable transient signals (col. 2, lines 20-21), and provides "low level software modules" including SCSI interface drivers within storage controllers to detect these transient signals (col. 3, lines 15-33). Upon detection of an error condition, test sequences are invoked, "at a low level of the RAID control module software (e.g., the driver level)," to determine the state of the disk drive (col. 3, lines 6-9). Upon the completion of these test sequences, failed I/O operations may be mapped to block addresses on a new drive, so that they may be retried (col. 7, lines 19-21). Thus, DeKoning teaches a method for recovering from disk drive insertion or removal based on data used by a control module (e.g., a SCSI interface driver) within a data storage system, not an identifier used by the operating system of a host computer to enable access to at least one computer system resource, as recited in claim 14. Indeed, the process of DeKoning is performed within a data storage system, based on information detected within the data storage system, by control modules within the data storage system. DeKoning simply does not disclose or suggest reconfiguring a computer system in response to a change to an identifier used by an operating system on the host to enable access to a resource.

Since neither Cannon nor DeKoning discloses reconfiguration in response to a change in an identifier used by a host operating system to access at least one system resource, the combination necessarily fails to show this limitation of claim 14. Thus, claim 14 patentably distinguishes over the asserted combination, such that the rejection of claim 14 under 35 U.S.C. §103(a) should be withdrawn.

Claims 15-20 depend from claim 14, and are allowable for at least the same reasons.

Claim 45 is directed to a computer-readable medium that, when executed on a host computer, performs a method substantially similar to that recited in claim 14. Therefore, for the reasons set forth above, claim 45 patentably distinguishes over the prior art of record, such that the rejection of claim 45 under 35 U.S.C. §103(a) should be withdrawn.

Claims 46-50 depend from claim 45, and are allowable for at least the same reasons.

Claim 72 is directed to a host computer comprising an operating system that uses at least one identifier to enable access to at least one computer system resource, and at least one controller to dynamically reconfigure the computer system in response to a change in configuration that changes the at least one identifier used by the operating system to enable access to the at least one computer system resource.

As should be clear from the discussion above in connection with claim 14, the prior art of record does not teach or suggest a host computer including at least one controller to dynamically reconfigure the host computer system to alter a manner in which an application program accesses at least one computer system resource in response to a change in configuration that results in a change in an identifier used by the operating system to access the computer system resource. Therefore, it is respectfully asserted that claim 72 patentably distinguishes over the asserted combination, such that the rejection of claim 72 under 35 U.S.C. §103(a) should be withdrawn.

Claims 73-77 depend from claim 72, and are allowable for at least the same reasons.

Rejections under 35 U.S.C. §103 Over Ma in view of Ha

Claims 1-20, 34-50 and 62-77 are rejected under 35 U.S.C. §103 over Ma in view of Ha. This rejection is respectfully traversed.

In the previous response (filed February 26, 2003), it was pointed out that the combination of Ma and Ha is improper, and that the claims patentably distinguish over the combination. Thus, for the reasons set forth in the previous response, which is incorporated herein by reference, the rejections of claims 1-20, 34-50 and 62-77 under 35 U.S.C. §103 over Ma in view of Ha are improper and should be withdrawn.

In addition, the Applicants note that the Office Action addresses the amendments to claims 14, 45 and 72 filed in the previous response by asserting that Ma "teaches identifiers to enable access to a resource" at col. 13-14, lines 62-23. However, as amended, claims 14, 45 and 72 recite, *inter alia*, dynamically reconfiguring a computer system in response to a change in a

configuration of the computer system that changes at least one identifier used by the operating system to enable access by the host computer to at least one computer system resource. In the cited passage, Ma discloses attributes of a class, wherein a class is a basic program unit used by a meta server object schema (col. 13, lines 40-50). Thus, the identifiers disclosed by Ma are those used by the meta server to define attributes of the class, not by the operating system to enable access by a host to a computer system resource, as recited in claims 14, 45 and 72. As a result, claims 14, 45 and 72 patentably distinguish over the prior art of record for this additional reason, such that the rejection of these claims and those depending therefrom under 35 U.S.C. §103(a) should be withdrawn.

Similarly, claims 1, 34 and 62 have been amended to recite a change in configuration that results in a change in an identifier used by a host operating system to access a system resource. Thus, these claims and those depending therefrom also distinguish over Ma and Ha for this additional reason.

CONCLUSION

In view of the foregoing amendments and remarks, this application should now be in condition for allowance. A notice to this effect is respectfully requested. If the Examiner believes, after this amendment, that the application is not in condition for allowance, the Examiner is requested to call the Applicants' attorney at the telephone number listed below to discuss any outstanding issues relating to the allowability of the application.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, that is not covered by an enclosed check, please charge any deficiency to Deposit Account No. 23/2825.

Respectfully submitted,
Steven M. Blumenau et al., Applicant(s)

By: _____

Richard F. Giunta, Reg. No. 36,149
Wolf, Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210-2211
Telephone: (617) 720-3500

Docket No. E00295.70087.US
Date: 08/27/03